65nm DC-Current-Free ReRAM Nonvolatile Computing-in-Memory Architecture
65奈米製程無直流電流之電阻式記憶體內運算架構

組別：A528　　指導教授：張孟凡　　組員：羅宇佑

Abstract

Resistive RAM (ReRAM) is a promising non-volatile memory due to its low power consumption, low operating voltage, and fast write speed. In paper[1], a novel architecture performs MAC (multiply-accumulate) operations by converting larger bit line (BL) current to lower discharge latency, avoiding direct current summation. Inspired by this idea, I implemented the design using a 65 nm CMOS process and conducted parameter tuning to optimize performance. During the process, I also proposed circuit-level modifications to reduce the impact of process variation, improving sensing accuracy in MAC operations.

Research Methodology

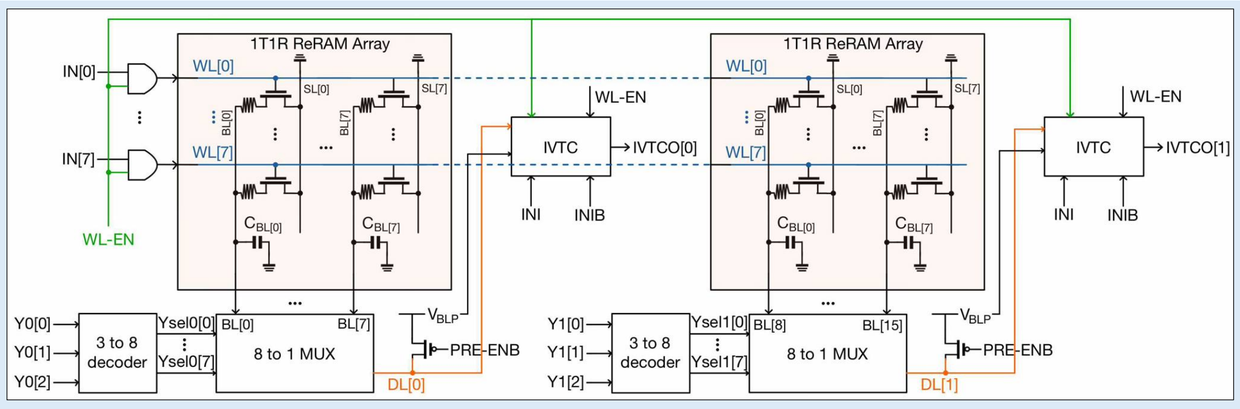

Fig. 1. Overall architecture with 2 IVTC cells

Eight 1-bit inputs are applied to the WLs of the ReRAM array. Each BL accumulates current based on input × weight across 8 cells, producing a MAC value from 0 to 8. Since MAC = 8 is rare, values 0~7 are mainly considered. To save area, one shared IVTC is used for all 8 BLs, as in paper [1].

In each cycle, a MUX selects one BL as the data line (DL). The DL is first pre-charged to V_{BLP}, then discharged through ReRAM cells. A higher MAC value results in more low-resistance cells (LRS), causing higher I_{BL} and faster discharge—leading to shorter discharge latency. This latency difference is small, so the IVTC amplifies it by converting the voltage change into a rising timing signal (IVTCO), as shown in Fig. 2.

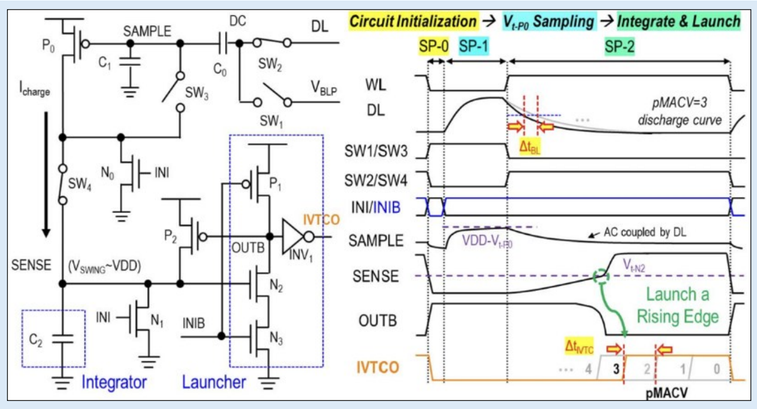

Fig. 2.
The structure and operation of IVTC (Integration-based Voltage-Time Converter) cell

The IVTC converts small DL voltage swings into a larger voltage at the SENSE node. As the BL discharges, ΔV_{DL} is coupled to the SAMPLE node, triggering a charging process that eventually discharges the OUTB node. A rising edge is then generated, amplifying the small BL discharge time difference (Δt_{BL}).

After simulation, I found that larger MAC values result in smaller IVTCO margins, so I focused on the critical case between MACV = 6 and 7. As shown in Fig. 1, two IVTC cells and MUXs are used to simulate and compare IVTCO[0] and IVTCO[1]. After some adjustment, Monte Carlo simulations are run to fine-tune parameters, aiming to enlarge the IVTCO margin and improve accuracy.

Experimental Results & Conclusion

After parameter tuning, the initial circuit had 433 failures out of 1024 Monte Carlo runs, resulting in 57.7% accuracy. I observed that the OUTB node was pre-charged to VDD but stayed idle before discharging, during which transistor N2 leaked current. Due to process variation in N2's threshold voltage (Vth), this leakage was inconsistent, reducing accuracy. To address this, I slightly modified the circuit so that OUTB remains at VDD until discharge begins, as shown in Fig. 4. This eliminated the idle leakage period and improved the Monte Carlo results, reducing failures to 395 and increasing accuracy to 61.4%.

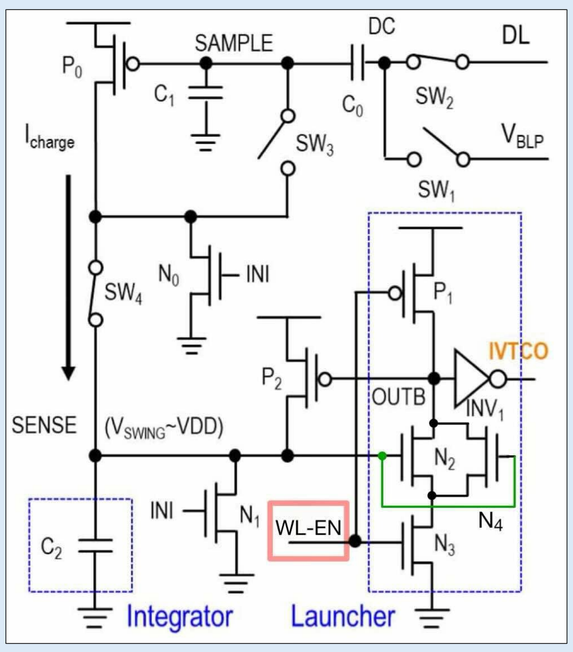

Fig. 3. The structure of IVTC_new cell

To further improve performance, I proposed a new version of the IVTC (IVTC_new) by adding a second transistor, N4, in parallel with N2. Both transistors were given smaller W/L ratios. Because of variation in their Vth values, N2 and N4 turn on at slightly different times, effectively averaging the activation point via positive feedback.

Statistically, if N2 and N4 each have $V_{th} \sim N(\mu, \sigma^2)$, their combined effective $V_{th}' \sim N\left(\mu, \frac{m^2+n^2}{(m+n)^2}\sigma^2\right)$. V_{th}' variation is more concentrated than V_{th}. This effectively reduces the impact of process variation on V_{th}.Monte Carlo results for IVTC_new showed further improvement, with only 361 failures out of 1024 runs, achieving an accuracy of 64.7%.

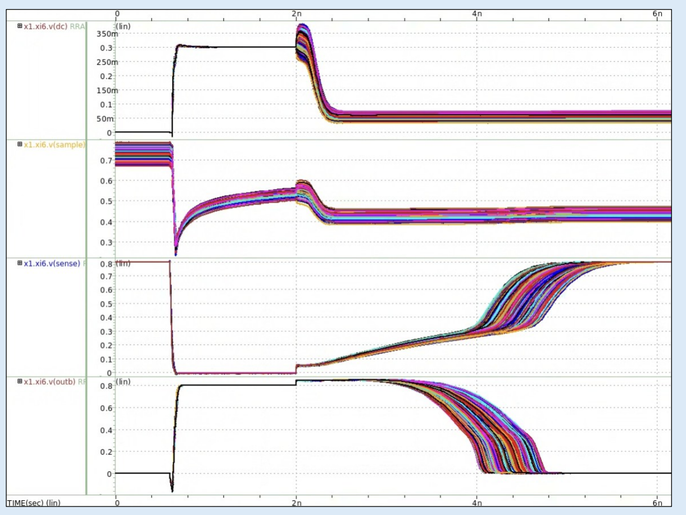

Fig. 4. Monte Carlo simulation for IVTC_new

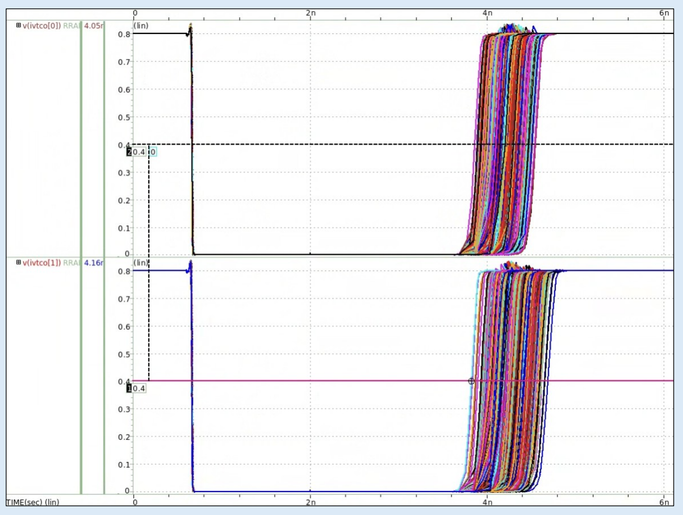

Fig. 5. Monte Carlo simulation of IVTCO[0] & IVTCO[1] for IVTC_new

Reference：
[1] J.-M. Hung et al., "An 8-Mb DC-current-free binary-to-8b precision ReRAM nonvolatile computing-in-memory macro using time-space readout with 1286.4-21.6TOPS/W for edge-AI devices," in Proc. IEEE ISSCC, 2022, pp. 182–184.
[2] C.-X. Xue et al., "24.1 a 1Mb multibit ReRAM computing-in-memory macro with 14.6 ns parallel MAC computing time for CNN based AI edge processors," in Proc. IEEE Int. Solid-State Circuits Conf.-(ISSCC), Piscataway, NJ, USA: IEEE Press, 2019, pp. 388–390.
[3] C.-X. Xue et al., "A 22nm 2Mb ReRAM compute-in-memory macro with 121–28TOPS/W for multibit MAC computing for tiny AI edge devices," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC), Jul. 2020, pp. 244–246.